Repeatability and Workability Evaluation of SIGMOD 2011

Philippe Bonnet[1], Stefan Manegold[2], Matias Bjørling[1], Wei Cao[3], Javier Gonzalez[1], Joel Granados[1], Nancy Hall[4], Stratos Idreos[2], Milena Ivanova[2], Ryan Johnson[5], David Koop[6], Tim Kraska[7], René Müller[8], Dan Olteanu[9], Paolo Papotti[10], Christine Reilly[11], Dimitris Tsirogiannis[12], Cong Yu[13], Juliana Freire[6], and Dennis Shasha[14]

[1]ITU, Denmark
[2]CWI, Netherlands
[3]Remnin University, China
[4]University of Wisconsin, USA
[5]University of Toronto, Canada
[6]University of Utah, USA
[7]UC Berkeley,USA
[8]IBM Almaden, USA
[9]Oxford University, UK
[10]Università Roma Tre, Italy
[11]University of Texas Pan Am, USA
[12]Microsoft, USA
[13]Google, USA
[14]New York University, USA

ABSTRACT

SIGMOD has offered, since 2008, to verify the experiments published in the papers accepted at the conference. This year, we have been in charge of reproducing the experiments provided by the authors (repeatability), and exploring changes to experiment parameters (workability). In this paper, we assess the SIGMOD repeatability process in terms of participation, review process and results. While the participation is stable in terms of number of submissions, we find this year a sharp contrast between the high participation from Asian authors and the low participation from American authors. We also find that most experiments are distributed as Linux packages accompanied by instructions on how to setup and run the experiments. We are still far from the vision of executable papers.

1. INTRODUCTION

The assessments of the repeatability process conducted in 2008 and 2009 pointed out several problems linked with reviewing experimental work [2, 3]. There are obvious barriers to sharing the data and software needed to repeat experiments (e.g., private data sets, IP/licensing issues, specific hardware). Setting up and running experiments requires a lot of time and work. Last but not least, repeating an experiment does not guarantee its correctness or relevance.

So, why bother? We think that the repeatability process is important because it is *good scientific practise*. To quote the guidelines for research integrity and good scientific practice adopted by ETH Zurich[1]: All steps in the treatment of primary data must be documented *in a form appropriate to the discipline in question* in such a way as to ensure that the results obtained from the primary data can be reproduced completely.

The repeatability process is based on the idea that in our discipline, the most appropriate way to document the treatment of primary data is to ensure that either (a) the computational processes that lead to the generation of primary data can be reproduced and/or (b) the computational processes that execute on primary data can be repeated and possibly extended. Obviously, the primary data obtained from a long measurement campaign cannot be reproduced. But our take is that the best way to document the treatment of these primary data is to publish the computational processes that have been used to derive relevant graphs. On the other hand, the primary data obtained when analyzing the performance of a self-contained software component should be reproducible. Ultimately, a reviewer or a reader should be able to re-execute and possibly modify the computational processes that led to a given graph. This vision of executable papers has been articulated in [1].

This year, as a first step towards executable papers, we encouraged SIGMOD authors to adhere to the fol-

[1]http://www.vpf.ethz.ch/services/researchethics/Broschure

lowing guidelines:[2]

(a) Use a virtual machine (VM) as the environment for experiments.

(b) Explicitly represent pre- and post-conditions for setup and execution tasks.

(c) Rely on a provenance-based workflow infrastructure to automate experimental setup and execution tasks.

Ideally, a common infrastructure guarantees the uniformity of representation across experiments so reviewers need not re-learn the experimental setup for each submission. The structure of workflows should help reviewers understand the design of the experiments as well as determine which portions of the code are accessible. While virtual machines ensure the portability of the experiments so reviewers need not worry about system inconsistencies, explicit pre- and post-conditions make it possible for reviewers to check the correctness of the experiment under the given conditions.

In the rest of the paper, we look back on the repeatability process conducted for SIGMOD 2011.

2. ASSESSMENT

2.1 Participation

Renée Miller, PC-chair for SIGMOD 2011, agreed to add a couple of questions to the submission site. 73% of the authors said that they would participate in the repeatability process. As we will see in Section 2.1.2, the percentage of accepted papers actually submitted to the repeatability and workability committee was limited to 35%. The reasons cited for not participating were:

1. intellectual property rights on software
2. sensitive data
3. specific hardware requirements

None of these reasons, however, explain the geographic distribution of authors participating to the repeatability process shown in Figure 1. This graph compares the number of papers accepted at SIGMOD and the number of papers participating in the repeatability process grouped by the region of origin of the first author (Asia, America, Europe, Oceania). While this grouping is largely arbitrary (some authors might not be associated to the same region as the first author), the trends that appears in Figure 1 is significant. To put it bluntly, almost all Asian authors participate in the repeatability process, while few American authors do. Some American authors have complained that the process requires too much work for the benefit derived [2], but we believe that several observations can improve this cost/benefit calculation

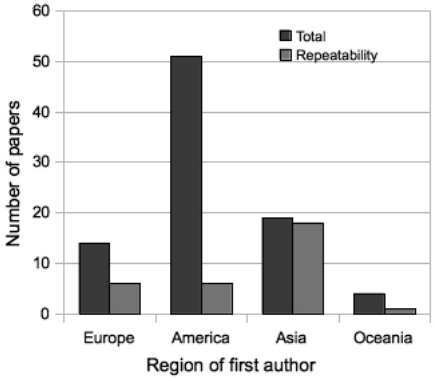

Figure 1: Distribution of participants to the repeatability process per region of first author.

1. **[more benefit]** repeatable and workable experiments bring several benefits to a research group besides an objective seal of quality: a) higher quality software resulting from the discipline of building repeatable code b) an improved ability to train newcomers to a project by having them "play with the system"
2. **[less cost]** using the right tools, a research group can make a research experiment repeatable easily (we are working on an upcoming companion article which contains a tutorial on how to make this happen).

2.1.1 Process

As in 2009, our goal was to complete the repeatability reviews before the conference started, so that authors could advertise their result during their presentation (a first straightforward way to guarantee some benefit for authors). We placed the submission to the repeatability committee at the same time as the deadline for the camera ready copy of the paper: leaving one month to the author of accepted papers to prepare their submission and leaving two months for reviewers to work on an average of three submissions each.

The availability of the Elastic Cloud Computing infrastructure via a grant from Amazon allowed us to experiment with a great variety of hardware and software platforms. Experiments were run on servers equipped with 26 CPUs or 40 GB of RAM, running OS ranging from Windows to CentOS. The availability of the Condor-based Batlab infrastructure from Miron Livny's group at U.Wisconsin allowed a reviewer to repeat a

[2]See the Repeatability section of the ACM SIGMOD 2011 home page: `http://www.sigmod2011.org/calls_papers_sigmod_research_repeatability.shtml`

cluster-based experiment with 40 machines - as opposed to 3 on the original paper. Note also that a few authors made their own cluster infrastructure available via a gateway which made it possible for reviewers to repeat the data acquisition phase of the authors' papers.

The most frequently asked question by authors at submission time was *where can I upload a package with the system and data needed to reproduce my experiments?*. Authors were asked to make their experiment available for download. This was a major problem for a Chinese group whose experiment package could not be downloaded properly despite numerous attempts. On the other hand, a group from ETH Zurich fully complied to ETH *Guidelines for research integrity* and made their software and data publicly available online[3].

A problem mentioned in the previous editions of the repeatability process was the high burden on reviewers when setting up experiments. To mitigate this problem, as explained in the introduction, we advocated this year that authors should consider submitting a virtual machine containing system and data. This effort was far from successful as illustrated in Figure 2. The vast majority of submissions were Linux or Windows packages with instructions on how to set them up and run the experiments. For most papers, the set up phase (specially on Linux) was well designed and required low overhead for the reviewer. However, many papers which did not get the repeatability label failed in the set-up phase, often because some dependencies had not been made explcit; such problems would have been avoided with a well tested virtual machine.

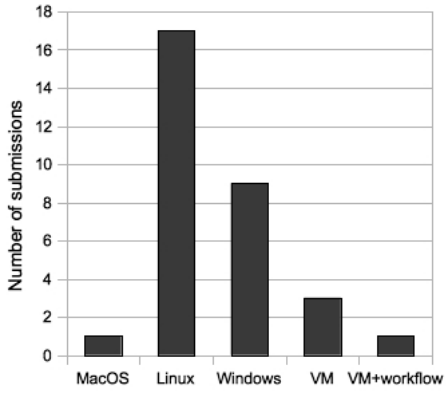

Figure 2: Operating system used for the submissions to the repeatability process in 2011.

Each paper was assigned a primary reviewer. A secondary reviewer was introduced in case the primary reviewers had problems downloading a submission, or setting it up because of OS or hardware mismatch. The load on the reviewers was quite uneven. Figure 3 shows the number of experiments per paper - which is a good indicator of the time needed to run the experiments. We still miss a good indicator for the time needed to setup the experiments. This year, we simplified the grades given to each paper: not repeatable, repeatable or repeatable&workable.

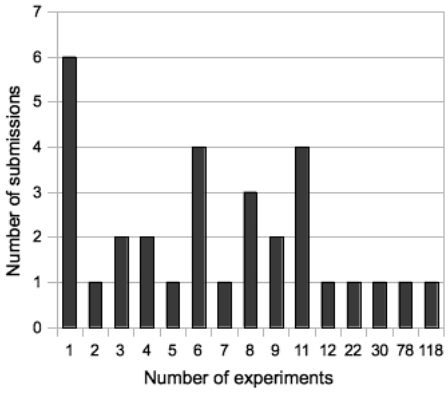

Figure 3: Distribution of number of experiments per submission.

This year, we set up a web site running on an EC2 server www.sigmod11repeatability.org with instructions for authors, a couple of examples showing how to use the Vistrails workflow engine to setup experiments and the submission site. We relied on an instance of HotCRP[4] to support submissions of instructions as well as anonymous interactions between authors and reviewers during the reviewing period. While HotCRP was fully satisfactory in terms of stability, functionality and ease of use; the setting of automatic emails from a GMail account created for the `sigmod11repeatability.org` domain turned out to be a problem - spam filters prevented mails and notifications sent by hotCRP to reach their destination.

2.1.2 Results

Figure 4 shows the results from the repeatability process since 2008[5]. In terms of percentages, the participation increased slighlty in 2011 compared to 2009 and 2010–those years where only accepted papers were considered for the repeatability process–while the percentage of repeatable papers remained stable.

The results were announced to the authors prior to the conference (at the exception of two papers). Results will be associated as labels on the existing article repositories

[3] `http://people.inf.ethz.ch/jteubner/publications/soccer-players/`

[4] `http://www.cs.ucla.edu/~kohler/hotcrp/`

[5] The results from 2008 and 2009 are presented in the SIGMOD Record articles [2, 3]; the results from 2010 are available at `http://event.cwi.nl/SIGMOD-RWE/2010/`

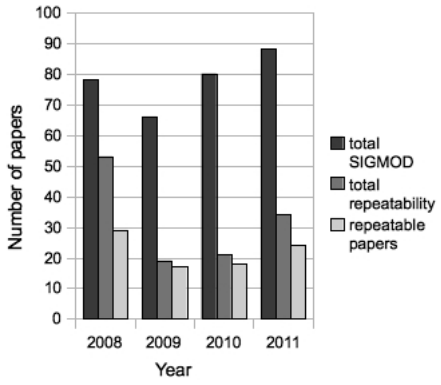

Figure 4: Repeatability results since 2008

(either ACM or PubZone). More importantly, the experiments themselves should be archived in a repository of repeatable experiment. Setting up such a repository for the SIGMOD community is the next obvious challenge.

3. CONCLUSION

The SIGMOD 2011 repeatability initiative attempted to increase participation and enhance the quality of submissions by offering tools to authors and reviewers. This has succeeded only partly: virtual machines and workflows simplify the process for reviewers but are harder to implement for authors than sending a shell script. Unfortunately, the shell scripts have many system dependencies that may make them difficult to repeat or to build upon by future researchers. An ongoing research challenge is to develop tools to help authors create high quality repeatable computational experiments with reasonable effort.

Acknowledgements

We would like to thank Amazon for their EC2 grant and Miron Livny and his team (especially Brooklin Gore) for their help with the Batlab infrastructure.

4. REFERENCES

[1] David Koop, Emanuele Santos, Phillip Mates, Huy T. Vo, Philippe Bonnet, Bela Bauer, Brigitte Surer, Matthias Troyer, Dean N. Williams, Joel E. Tohline, Juliana Freire, and Cláudio T. Silva. A provenance-based infrastructure to support the life cycle of executable papers. *Procedia CS*, 4:648–657, 2011.

[2] S. Manegold, I. Manolescu, L. Afanasiev, J. Feng, G. Gou, M. Hadjieleftheriou, S. Harizopoulos, P. Kalnis, K. Karanasos, D. Laurent, M. Lupu, N. Onose, C. Ré, V. Sans, P. Senellart, T. Wu, and D. Shasha. Repeatability & workability evaluation of SIGMOD 2009. *SIGMOD Rec.*, 38:40–43, December 2010.

[3] I. Manolescu, L. Afanasiev, A. Arion, J. Dittrich, S. Manegold, N. Polyzotis, K. Schnaitter, P. Senellart, S. Zoupanos, and D. Shasha. The repeatability experiment of SIGMOD 2008. *SIGMOD Rec.*, 37:39–45, March 2008.